Location Privacy without Carrier Cooperation

Keen Sung Brian Neil Levine Marc Liberatore
School of Computer Science, Univ. of Massachusetts Amherst
{ksung,brian,liberato}@cs.umass.edu

***Abstract*—Cellular network operators can track the location of cell phone users as they connect to different towers. Operators may not directly control the user's phone, but they do supply and control the SIM card that identifies the user. We seek to preserve a cellular phone user's location privacy from cellular network operators. We propose the ZipPhone protocol for secure, virtual, and therefore easily changeable SIM cards. ZipPhone breaks the association between the user and IMSI identifier, and thus prevents the cellular operator from localizing the user. At the same time, it still allows authentication, billing, and E911 service by the operator. We empirically analyze the effectiveness of ZipPhone against a *passive* carrier. This class of attacker has a location profile of the user before they switched to ZipPhone, but relies on the normal operation of GSM mechanisms to learn the location of users. We reproduce the results of a previous inference study and show that it did not realistically model GSM carriers. We show that ZipPhone users can expect to be deanonymized only 6% of the time, which is a sixth of the rate reported by previous work.**

I. Introduction

Mobile device users depend on the cellular phone and data network infrastructure on a continuous basis. Cellular network operators can track cell phone users as they travel among and connect to towers, violating their location privacy. In this work, we propose a scheme to preserve users' location privacy by allowing them to easily change virtual SIM cards over the Internet. This scheme, which we call *ZipPhone*[1], is backwards-compatible with existing cellular infrastructure.

The difficulty of preserving the location privacy of cellphone users is that an entire infrastructure of densely placed radio towers can observe the phone's signals. And although the cellular carrier doesn't necessarily control the phone hardware, they do supply and control the SIM card carried by the user in their phone. SIM cards contain a small computer, protected by tamper-resistant packaging, able to process cryptographic functions and carrying a unique identifier. Despite this scenario, we believe location privacy is still possible.

We detail a method for obtaining location privacy without the active cooperation of the carriers that control the cellular infrastructure. Rather than obfuscating the location of the user, our approach is to break the one-to-one link between user and SIM card, as the latter is what carriers use to identify users of localized handsets. As we will demonstrate, ZipPhone can provide location privacy from current carriers because their defenses are aimed at mitigating *cloning*, which is the unauthorized duplication of the credentials stored in the SIM card; ZipPhone lends credentials rather than cloning them. If carriers targeted their defenses on ZipPhone the problem would be harder, since they might deny service to phones that appear to be running ZipPhone, or they might localize phones beyond normal GSM operation.

At a high level, ZipPhone works as follows. ZipPhones do not have a SIM card. Instead, they access a remote set of credentials to instantiate a virtual SIM locally. The credentials are accessed using Wi-Fi or other networks not observable to the carrier. Once a virtual SIM is obtained, the usual GSM protocols are still able to handle switching among towers, refreshing virtual SIMs, and other details.

In sum, our contributions are as follows.

- We propose a backwards-compatible method of GSM location privacy that does not rely on the cooperation of the carrier. New credentials can be re-issued relatively frequently over the air. Our primary design relies on a cooperative but untrusted *mobile virtual network operator* (MVNO) that issues ephemeral identities and session keys. We also propose a variant of this system, which uses *peer-based* exchange of SIM credentials, and relies on flaws in the GSM system first discovered in 2001 and never fixed. We call this variant *PeerPhone.*
- We propose a method of thwarting a class of active attacks on users that reveal their location. Such attacks are generally based on GSM pages for fake phone calls or SMS messages. Our solution is an application of portknocking [34].
- We empirically analyze the effectiveness of ZipPhone against a *passive* carrier. This class of attacker has a location profile of the user before they switched to ZipPhone, but relies on the normal operation of GSM mechanisms to learn the location of users. We reproduce the "conclusive" results of a previous inference study [41] and show that it does not realistically model GSM carriers. We show that ZipPhone users can expect to be denonymized only 6% of the time, which is less than a sixth of the rate reported by previous work.
- We discuss other attacks against ZipPhone. First, we detail how carriers can determine that a phone is using ZipPhone and therefore deny it service. Second, we discuss *active* attacks based on localizing the phone without their cooperation (e.g., lateration attacks).

In the next two sections, we review the operation of GSM networks and define our security model. We then present the details of ZipPhone, discuss its validity in terms of carrier terms-of-service, analyze its security, and place it in the context of related work.

[1]Our system is named for its parallels to the Zipcar rental system. In ZipPhone, the user doesn't own the SIM, she rents it only temporarily as she moves around town.

II. Summary of GSM Networks

Cellular carriers use one of a few technologies to support mobile phones. In this section, we give an overview of the details of the GSM system and terminology [54], and the aspects of the GSM architecture that are required for our discussion of ZipPhone. ITM-2000 based systems have differences but largely follow the same general architecture; GSM is the foundation of GPRS, WCDMA, EDGE, UMTS, and LTE. Moreover, most networks support reverting to a GSM mode.

A phone handset is known in GSM as a *mobile station* (MS), and each is composed of its *mobile equipment* (ME) and its *subscriber identity module* (SIM) card. To connect, the handset connects via its radio interface to a radio tower, which is called a *base transceiver station* (BTS). A single *base station controller* (BSC) can manage many BTSs, including functions related to resource and mobility management. The BTS and BSC form the *base station subsystem* (BSS).

A. Connecting to the Network

Each BSC is connected to one *mobile switching center/visitor location register* (MSC/VLR); see Fig. 1. The MSC/VLR controls call setup and routing, among other tasks. One MSC/VLR connects to many BSCs. Each MSC/VLR is also connected to the carrier network's *home location register* (HLR), which records the particular MSC/VLR where each phone subscriber may be found. The HLR is associated with an *authentication center* (AuC) that stores cryptographic credentials needed for communicating with each of the carrier's SIM cards.

Each SIM card contains an *international mobile subscriber identity* (IMSI) and a unique symmetric key K_i. The same IMSI and key pairing is stored in the AuC. Generally, SIMs do not allow querying of the key. Each ME contains an *international mobile equipment identity* (IMEI) number, that is comprised of a unique serial number and the make, model, and place of manufacture of the phone.

When a phone is powered on, it scans for BTS towers available from the carrier specified in the SIM, each on a different frequency. To connect to the network, the phone begins a *location update* process by selecting the tower with the strongest signal and then requesting a communications channel from the BTS. As part of the process, the handset sends its IMSI to the MSC via the BTS. (If requested, the handset will also send its IMEI.) The MSC will request authentication information from the HLR/AuC with a *mobile application part* (MAP) *send authentication info* message for this particular IMSI. MAP is the internal signaling protocol for GSM nodes.

To generate a response, the HLR/AuC selects a *random number* and combines it with the IMSI's K_i to generate a *session key* (K_c) and a *signed response* (SRES), using a given algorithm. The three values (called a *triplet*) are returned to the MSC/VLR. The MSC forwards, via the BTC, only the random number to the handset. The handset asks its SIM to generate the signed response that corresponds to the random number (for the given algorithm), and forwards the value to the MSC/VLR via the BTS. The SIM also generates K_c for the phone from the random number.

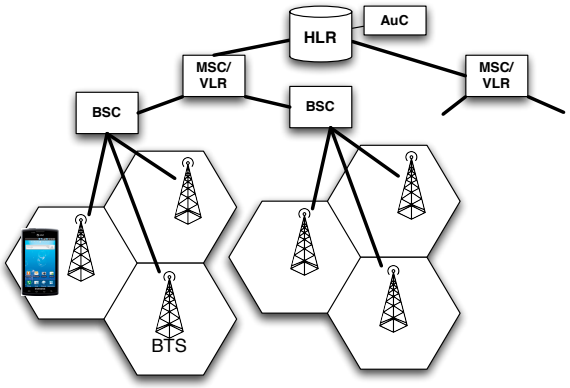

Fig. 1: The simplified GSM architecture.

If the value matches, the handset is authenticated, and the MSC sends a MAP *update location* message to the HLR. The HLR associates the IMSI with the address of the MSC/VLR. The MSC then tells the BTS the session key, and then the phone and BTS switch to an encrypted channel. The MSC assigns and sends the phone a *temporary mobile subscriber identity* (TMSI) that is unique to the *location area* it resides in. Although the TMSI is sent encrypted, it can be retrieved from the SIM on some phones and there are scenarios where it is broadcast over the network unencrypted [35]. Finally, the phone releases the communications channel. It is now *camped* on a cell and is ready to use the network's services. For example, incoming calls from the public switched telephone network are routed through the HLR, then the MSC/VLR, and after a broadcast *page*, from the BTS to the handset.

B. Moving to a new location area

As the user moves within the cells that comprise a location area, it can use the same TMSI to access the network. Once the handset moves to a new location area, it must begin the location updating process again, though it can send its TMSI (and old location area) instead of the IMSI. If the TMSI value is already known to the MSC/VLR, the location update will be accepted without re-authentication. A new TMSI will be assigned, unique to the new location area.

If the location area was managed by a different MSC/VLR, then the handset's credentials must be reacquired. It is preferred for the MSC to request the associated IMSI from the previous MSC/VLR, rather than directly from the handset. The new MSC/VLR also retrieves a triplet from the HLR and may ask the phone to re-authenticate with a new signed response.

C. Data access

GPRS provides data service for mobile phones, and it is the basis for UMTS architectures [3]. The protocol for activating data service is called a *GPRS attach*. The process is similar to a location update, and we only sketch the details. After requesting and receiving a channel, the MS sends an attach request to its BSS. The BSS forwards the request to a *serving GPRS support node* (SGSN), which fetches authentication triplets from the HLR. Once the

MS authenticates itself based on knowledge of K_i, an encrypted channel is started between the SGSN and the mobile. The SGSN exchanges information with the HLR about the authenticated handset, and the attach is accepted and acknowledged to the handset, including issuance of a *packet temporary subscriber identity* (P-TMSI). The process will be repeated if the handset moves to a location area controlled by a different SGSN. UMTS data access is essentially the same in terms of the signaling to the handset and the authentication mechanisms.

III. Security Model

Our security model consists of users seeking location privacy from cellular carriers. Users carry and control phones that are equipped with cellular and Wi-Fi radios, and thus have IP-based data connections on both radios (once connected). They are allowed to use encryption, which we assume is reasonably strong. Users are not trying to gain unauthorized service.

Our model includes carriers that are *passive* or *active* attackers.

Passive Attacker. A passive carrier controls the network infrastructure, accounts, and SIM cards. They can ensure that only authorized accounts are used to connect to the network. They know the incoming and outgoing calls of each phone and can observe data packets sent via the phone's cellular connection (but not the contents of encrypted traffic). The passive attacker has records of which towers a user's phone has associated with during voice and data transfers, and knows the location of each of its towers and its coverage area. The attacker cannot force the phone to answer requests (e.g., pages). The carrier and SIM cards make use of GSM protocols including the A5 family of ciphers and their existing weaknesses. We also allow the passive attacker to leverage past information about user geographic movements to build a profile.

Networks with an *always-update* policy expect phones to perform a location update whenever the phone enters a new cell; operators never enforce such a policy because it has the highest overhead for the carrier [48], [61]. The standard is for phones to perform a *forming location area (LA) update* [48], [61], where a phone initiates a location update only if its location area changes. In that case, the carrier will learn the exact cell that the phone is associated with only when it *(i)* performs the location update or GPRS attach, *(ii)* connects to a BTS to make a call or send data, and *(iii)* answers pages for incoming calls or data. The passive attacker does not falsely or unnecessarily prod the user into performing a location update or similar attacks.

Active Attacker. The active carrier has all the abilities of the passive attacker and can attempt real-time geolocation a specific phone, including unnecessarily asking the phone to initiate a location update process and multilateration of the received phone signal.

We do not consider certain other active attacks due to either their fragility or their extreme cost when applied on a network-wide scale. For example, we do not allow either attacker to use cameras or other visual information, nor to physically stalk the user. Because the carrier does not control the phone hardware or OS in our model, only the SIM, it cannot, for example, insert SSL keys onto the phone or otherwise change the phone's software. Similarly, the carrier cannot prevent Wi-Fi connections, nor capture information from the phone's Wi-Fi, key presses, camera, GPS, or screen.

Handset signatures. Phone hardware has been shown to carry many unique signatures that can be determined remotely, including the radio's power amplifiers [45], [46] and the phone's accelerometer [15]. Although these attacks can be effective, they are not the focus of this paper. To obtain privacy, users need *both* the protocol we propose presently, and defenses against remote inference of physical signatures.

IV. ZipPhone

Cellular users are subject to several vulnerabilities that reveal their geographic position, though some are easily mitigated. First, phones store a unique IMEI, which is akin to a MAC address; however, this identifier can be modified by the user since she controls the handset hardware[2]. Second, each user is likely identifiable by the unique set of outgoing calls they make; however, they can make calls via VOIP rather than using the cellular carrier. Encryption of the VOIP stream can thwart carrier eavesdropping. Stronger protection is available by using VOIP over Tor[3]. Third, as our past work has shown, their location may been inferred from throughput characteristics, [55], though such attacks are limited, currently.

A naive privacy solution is offered by some *mobile virtual network operators* (MVNOs), such as TracPhone, Straight Talk, and Boost Mobile. MVNOs have no cell tower infrastructure. Instead, they buy service wholesale from carriers and resell to customers. Many MVNOs allow users to purchase a so-called "burner phone" and a SIM with pre-paid minutes, which can be disposed of by the user; such purchases and activations do not require a name or address. This approach to privacy works well for those willing to sacrifice a great deal of convenience and money for privacy, but it does not scale to improving the location privacy of every day cellular users that are seeking not to be "observed in all matters" [51].

In the remainder of this section, we identify vulnerabilities that stem from the use of a specific IMSI (stored on a tamper-proof SIM) and the geographic consistency of IMSIs (which in practice are typically tied to a home, workplace, or both). We then detail our approach to mitigating these vulnerabilities.

A. Authentication and Location Updating

The most naive approach to obfuscating the relationship between a user and a geographic trace of a phone by a cellular network is purchasing a pre-paid SIM card from an MVNO such as TracPhone, and replacing both it and the phone after a short period of time. A slightly better approach is for users to meet up in a cafe and randomly switch SIM cards and phones. Of course, neither approach is remotely

[2]See https://www.blackphone.ch/ for a recent attempt at a handset designed around consumer privacy.

[3]See http://torfone.org/

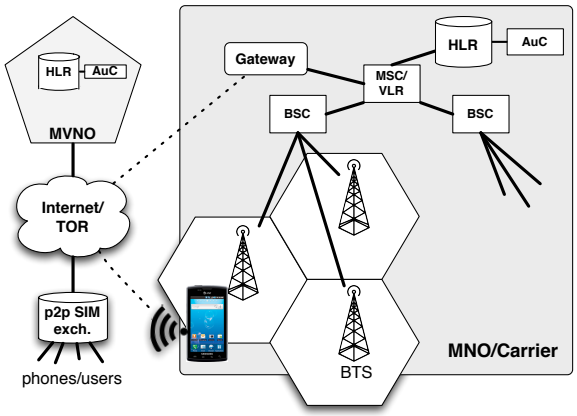

Fig. 2: The ZipPhone architecture for authentication and attachment to the network.

convenient. Further, such a scheme is not secure unless we can quantify how often a user should swap hardware.

ZipPhone allows for convenient exchange of logical cellular identifiers. It is based on phones without local SIM cards; instead, the phone's software retrieves a *virtual SIM* offered by Internet-accessible third party; see Fig. 2.

Carriers that control infrastructure, such as Verizon and AT&T, are generally called *mobile network operators* (MNOs). They market and sell service directly to consumers. MNOs also have a large *wholesale* business servicing MVNOs, which are explicitly allowed to resell service from an MNO. An MVNO has no national BSS infrastructure, but may issue SIMs and operate an HLR and AuC. About 100 MVNOs operate in the US alone [60].

ZipPhone is based on a *centralized* MVNO solution, though the operator is not trusted with identities. Below, we also detail a variant called PeerPhone which is decentralized. ZipPhone is a more secure solution than PeerPhone, but the former has the disadvantage that a business relationship with a major carrier is required.

Authentication. Any existing or new MVNO that operates an HLR and AuC could easily support the ZipPhone architecture. We assume that such an organization is cooperative (it does not launch denial-of-service attacks) but is not trusted by the ZipPhone users with their locations or identities.

Let's call one such MVNO *Zipline*. Like any MNVO today, Zipline customers roam on the network of a partnering MNO. There are two key differences between Zipline and current MVNOs on the market.

First, Zipline customers are not issued physical SIM cards; instead they anonymously request ephemeral IMSIs and IMEIs from Zipline's SIM Exchange during bootstrap (e.g., via Wi-Fi). IMSIs are ideally issued as nonces, but could be reused over time. IMSI requests are paid for (over Tor) via a suitably anonymous currency such as Bitcoin [42] or Zerocoin [39] in which no consistent ID is needed. Using Tor increases the difficulty of a SIM holder inferring that two ZipPhone requests originated from the same user. We can assume the leases are for some number of minutes at a time. The MVNO will know when the lease is initiated because it will receive an authentication request from the MNO. To terminate the lease, Zipline can send a message to the MNO's VLR, as per current GSM standards that exist to support the existing pre-paid plans of MVNOs.

Second, Zipline customers *prefetch* the correct answer to the next several SRES values. SRES prefetching has two key advantages: *(i)* it removes any long delays during signaling that identifies the phone as participating in ZipPhone, preventing expiration of GSM timer T3260 [2] and reducing the possibility of detection by the carrier; *(ii)* it ensures that the phone can perform location updates while in the field without a separate Internet connection. New purchases and prefetched triplets can be made anonymously over the data channel even if out of range of the initial Wi-Fi base station.

No aspect of Zipline requires any changes to the GSM architecture. While it does require cooperation of a sponsoring MNO, what we propose does not violate the terms of wholesale service of major carriers to their MVNOs. It is not necessary for the MNO to know that the MVNO is running ZipPhone. It is not out of the ordinary for an MVNO to run its own HLR/AuC. If a current MVNO implemented this architecture, a partnering MNO would be unable to easily distinguish customers with physical SIMs from those leasing remote SIMs.

Location Updates. Conveniently, as the user moves to new BTS towers and even new location areas, the TMSI will be refreshed without re-authenticating, as noted in Section II-B. The carrier will require the phone to re-authenticate only when a location area operated by a new MSC or SGSN is reached. These regions can be large, as cells range from about 1 to 100 km^2, one BTS manages several cells, and one MSC (or SGSN) manages several location areas. When re-authentication is required, ZipPhone users use prefetched values.

To make and receive phone calls, we assume the ZipPhone user has registered with an (anonymous) VOIP service, as described in Section IV-C. A significant limitation of ZipPhone is that the user cannot utilize the legacy phone system and reasonably expect to retain location privacy. However, E911 service, which is tied to a handset and not a user or SIM, would be still available if needed.

B. PeerPhone: Peer-based exchange of SIM credentials

To our knowledge, no current US-based MNO or MVNO allows the resale of their services by their end-users[4]. Customers of these operators may still wish to obfuscate their movement by occasionally *lending* their SIM to others for use without recompense. We call this variant scheme *PeerPhone*, and differentiate it from ZipPhone as needed in the remainder of this paper.

Lending an authorized phone and SIM, virtually or physically, does not appear to be against the Terms of Service of major carriers. This lending makes sense only for persons with unlimited data and call plans.

Authentication. A user that has been lent a SIM must be able to answer the challenge-and-response authentication

[4]Verizon's consumer Terms of Service: http://www.verizonwireless.com/b2c/support/customer-agreement. AT&T's: http://www.att.com/shop/en/legalterms.html?toskey=wirelessCustomerAgreement; T-Mobile's: http://www.t-mobile.com/templates/popup.aspx?PAsset=Ftr_Ftr_TermsAndConditions&print=true.

protocol posed by the carrier: PeerPhone relays the challenge to the SIM owner over Wi-Fi. This approach is successful because the GSM SIM security protocols require and ensure only that the SIM is authorized to use the network (as described in Section II-A), rather than requiring it is physically connected to the handset. The remote SIM holder is able to provide session key K_c and the SRES responses to the phone. Because K_i does not need to be passed, the SIM's use can subsequently be lent out to another user.

Relaying the carrier's packets across the Internet via Tor to the SIM holder will introduce significant delays. However, because carriers expect to work with resource-poor phones and do not wish customers to find it hard to connect to the network, the GSM protocol is not stringent. The GSM specifications set a 12-second timeout by default on receiving a response to authentication requests within a location update process (GSM timer T3260 [2]). If the timeout occurs, the user can simply try again.

The termination of the remote use of a SIM (and its IMSI) is easily controlled by the SIM owner, who need only connect to the carrier and perform a location update. In doing so, the HLR of the carrier will remove access from the previous or current MSC/VLR. A SIM owner can leave an extra Internet-connected phone at their house or other preferred location (perhaps with the actual SIM) and instruct it to connect to the network remotely; this setup allows the owner to reclaim without necessarily revealing their current location.

Retrieving Credentials from a SIM. Users that wish to participate in PeerPhone and lend out use of their SIM must retrieve the K_i key stored in it, though it only needs to be done once. They can then produce the K_c and SRES values for a remote PeerPhone requester, who can relay via Wi-Fi (or existing cellular connection) the *random number* issued by the carrier to the peer during a location update. Keys for encryption with GPRS/EDGE are also based on knowledge of K_i and can be similarly relayed. The peer should never relay the value K_i.

To recover K_i, peers can launch one of many known plaintext- and ciphertext-only attacks [5], [10], [47] against the various A5 encryption algorithms used by carriers. Barkan et al. [5] show how to recover the key in seconds from small amounts of ciphertext for both voice and GPRS communication for A5/1, A5/2, and A5/3. Almost all networks use A5/1 or no encryption at all (which is called *A5/0*) [52], [56]. (A5/2 is so weak that it is no longer allowed on GSM networks.)

All communications and all algorithms on a GSM phone are based on K_i; as long as carriers allow any portion of their communications using A5/1, the peer can recover the key by *bidding down* from the stronger A5/3 algorithm to A5/1 during negotiation (which is the basis of many commercial surveillance products [4]). In December 2013, Deutsche Telekom upgraded from A5/1 to A5/3, but still allows breakable connections via A5/1 from phones [40], and to date it has announced no changes to its US-based operation, T-Mobile. AT&T has announced it will upgrade to "parts" of its network, but because it will still allow A5/1 connections, peers can recover their K_i values [56].

By handing out K_c, the remote PeerPhone user can use the same attacks to recover K_i, but that vulnerability is already exploitable by all attackers that are within radio range of the SIM holder; it is no weaker to lend K_c to remote parties than it is to use the carrier's network in the first place. A5/1 was first shown to be weak in 2001 [10] — more than a decade ago — but this revelation has resulted in no deployment changes by carriers. Notably, smart phones on the market are close to having more resources than the PCs first used to carry out these attacks in 2001 (128 MB RAM and 146 GB storage [10]).

Location Updates. A PeerPhone can use its TMSI values to move among BTSs and location areas. When a location area is reached that is controlled by a different MSC or SGSN, the phone must reauthenticate. In that case, the PeerPhone must connect over the existing cellular data connection to the SIM holder. In some cases, the PeerPhone will be able to use its existing connection on the old tower to relay packets back to the SIM holder. To do so, it will need to switch its frequency several times to act as a relay; most phones can do so quickly as they are constantly switching frequencies to test received signal strength (to determine if a handoff is necessary, even during a voice call). If this back-and-forth cannot be completed reliably in practice, a PeerPhone can be engineered to consist of better antennas to allow more time for the relayed location update process to take place. In the worst case, a phone can be built with two radios to avoid the delay of frequency switching. In this preliminary work, we have not evaluated the efficacy of these mechanisms.

C. Preserving Privacy During Communication

By initiating and receiving overt GSM or unencrypted VOIP calls, ZipPhone users risk being identified via a profile of call records held by the carrier. Some protection is gained from using an encrypted VOIP service since it would not reveal to the carrier the identity of the user, whom she calls, or from whom she receives calls. If the VOIP service itself cannot be trusted, then an anonymous VOIP service is required, such as Torfone. Anonymous VOIP has a performance penalty [36].

An additional threat is posed by several attacks that request the user's phone to silently associate with a real or unauthenticated tower [22] based on an SMS Class 0 message, ICMP ping, or similar technique. The general approach of these attacks is to page the phone, and once the phone is associated to a cell, the call is abandoned. It's easy for ZipPhones to ignore all incoming calls and SMS messages, but since legitimate incoming VOIP calls are first received as pages for incoming data packets, a similar attack exists based on pages for GPRS data.

Thwarting Fake Cellular Pages. We propose an application of portknocking [14], [34] to defend against these attacks. The defense, which we call *pageknocking* has the limitation that it is easily subject to a denial of service attack by the carrier, and the user can't deny that it is using the defense (and therefore also using ZipPhone).

We assume that the user is registered to an Internet-based proxy service that accepts incoming VOIP calls and anonymously forwards the calls to the user. (We assume the registration process also creates the necessary entry in the carrier NAT table.) To signal to the user that an incoming

packet is genuine, the service uses a notification coded in a series of intervals between pages. More generally, this defense can be used for all incoming GPRS traffic to the phone if a proxy, VPN concentrator, or Tor node understands the protocol. In fact, a limitation of the approach is that, because unrelated incoming traffic can corrupt the signal, all traffic to the peer should be routed through one Internet-based proxy (or Tor-based service).

The phone waits for incoming VOIP calls via pages on the GPRS tunneling protocol (GTP). The phone waits in standby mode, which allows the carrier to know its location area for routing data, but not the specific cell nearest to the phone. Ahead of time, the service and the user agree upon a key and a parameter n. The key can be renegotiated at any time but we refer to the current key as g. A sequence of n bits is chosen as the first n bits of the *hmac* [6] of the key and the current time t, synchronized to the current minute: $hmac(g, t)$. The bit string determines a sequence of transmits and pauses. For each bit that is set, the service transmits a data packet, which results in a GTP page to the phone, and then the service pauses for a duration of d seconds. For each bit that is clear, a packet is not set and the service pauses for d seconds. The duration of d must be long enough to ensure that a new page will be generated by the carrier when a new packet comes in, but not so long that the chances of another packet being received is significant.

The chances that the carrier (or non-malicious third party) can generate a false n-bit page is 2^{-n}. At the receiver end, when a page is received at time t', the user calculates the first n bits of $hmac(g, t')$, and holds t' constant until the pattern doesn't match. If they do match, the phone responds to the page.

The value of d limits the number of incoming VOIP calls (and data packets generally) that a user can receive per minute. However, since the pattern changes once per minute, the number of chances that a third party has to brute force the value g is extremely limited. The protocol does not limit the amount of communication that the user can initiate.

V. Passive Attack Analysis

In this section, we characterize the effectiveness of the passive attacker against ZipPhone (and PeerPhone). We ask, *What percentage of the time can the passive attackers determine the identity of a ZipPhone user, given a profile of that user trained using overt data?*

Past work is pessimistic about the success of methods for location privacy that are based on changing identifiers. For example, Mulder et al. [41] previously examined the accuracy of techniques that infer the identity of a cellular user given a month-long training period and varying frequency of changing IDs. In that paper, they expected that IDs could be changed as infrequently as once a month or as often as once an hour. Attacker identification rates were as high as 88% for monthly changes and 48% for hourly changes; the paper remarks that "this work conclusively demonstrates that removing identifiers from location information, or merely blurring the spacial resolution, does not eliminate the danger of deanonymization." Below, we reproduce their classification method and evaluated the same data set [17], and then perform a critical additional experiment. Mulder et al. assume an *always-update* location management policy in which the user performs a location update with the carrier for *every* cell it enters.

As we show, a more realistic model of carrier location management based *forming LA updates* (see Section III) predicts such identification attacks are much less successful than previously reported. For example, when users change identifiers once an hour, attackers succeed only 6% of the time.

The code for our simulations is available from http://traces.cs.umass.edu.

A. Empirical Analysis of Passive Attackers

Given the ubiquity of cell phones, it is reasonable to expect the carrier to already have a profile of its users. Therefore, our evaluation of ZipPhone in the presence of a passive attacker assumes that the training data is easily accessible. We cross-validated our results using three test/train month pairs, encompassing the majority of the Eagle et al. [17] data set. The data includes the location areas, cells, and a record of calls made by 106 users between November 2004 and February 2005.

Because ZipPhone allows for users to obtain new IMSI values over the air, it's reasonable to expect that they can change identifiers fairly frequently. We have not implemented ZipPhone, and so we cannot guarantee any particular frequency. But for the sake of comparison, we show results for a frequency as (impractically) low as once a minute.

Inference Algorithm [41]. The inference approach designed by Mulder et al. is summarized as follows. During training, a square transition matrix P_k is created for each user k. For a set of all states S visited by all users, the matrix encodes the probabilities that a user in cell p moves to cell q for all $p, q \in S$.

$$Pr(k_{p,q}) = \frac{Count(p \to q)}{\sum_{q' \in S \setminus p} Count(p \to q')} \tag{1}$$

Self-transitions are ignored (i.e, $Pr(k_{p,p}) = 0$). We include a smoothing factor by adding 1 to each of the counts of all possible transitions before computing Eq. 1.

During testing, a window of M minutes of data from a single user is randomly selected from the held-out test data set. A contiguous sequence of cells $\theta = x_1, x_2, \ldots, x_n; x_i \in S$, consisting of n cells visited by an unknown user, is taken in chronological order from within that window. The sequence of $n-1$ transitions is evaluated against the values stored in the matrix for each user. An indicator is defined for each user k as the product of all probabilities in the associated training matrix. The user with the maximum value of I_k is selected by the classifier.

$$\arg\max_k I_k = \prod_{i=1}^{n-1} Pr(k_{x_i, x_{i+1}}) \tag{2}$$

While this model does not consider stationary location data (i.e. self-transitions), Mulder et al. found this approach more successful than a more sophisticated Markovian model that we did not reproduce.

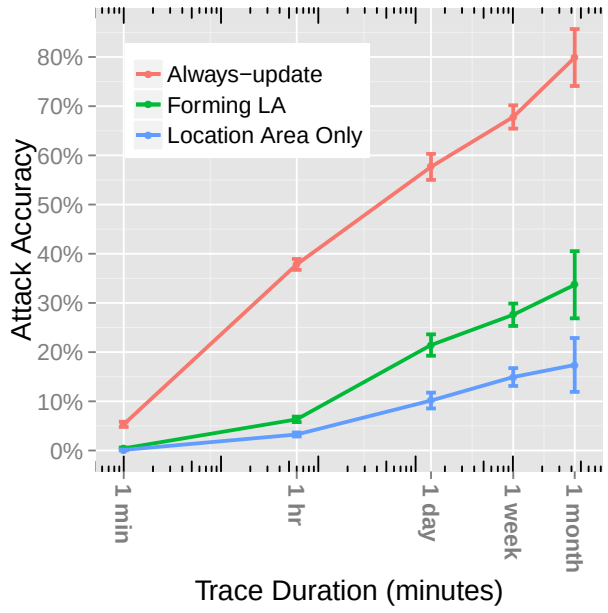

Fig. 3: The accuracy of the attack defined by Mulder et al. [41] under the always-update policy (top, red line) and forming LA policy (middle, green line). Our results match well with [41]: an attacker achieves a 38% success rate against users that update their SIM-based identifiers once an hour. Under the latter, more realistic, forming location area update policy, the attacker's success rate falls to 6% when SIM-based identifiers are updated once an hour. The bottom, blue line shows the lower bound on any scheme: it represents an unrealistic location management scheme where the carrier learns only the location area but not the cell a user is associated with. Errorbars represent 95% c.i.

Results. The effectiveness of the classifier is shown in Figure 3. In all cases, the attacker is given a preceding month's data as ground truth for training. The top, red line is a recreation of results from Mulder et al.: a randomly selected 1-month-long sequence of the cells a user is associated with results in a high accuracy of 80%; Mulder et al. saw[5] about 82%. A random sample of up to 1-hour of cell locations is identifiable 38% of the time; Mulder et al. saw about 44%. In both cases, random chance would be correct about 1% of the time.

Mulder et al.'s results are pessimistic in that they are for an *always-update* scenario, which is never implemented by carriers in practice [48], [61]. The attacker's accuracy is much lower for a more realistic *forming update* policy, where the carrier only learns the exact cell that the phone has associated with when the mobile device performs a location update — the device performs an update only upon entering a new location area or receiving a call (rather than upon entering each cell within a location area), or when the carrier initiates a connection. Fortunately, Eagle et al.'s data includes a record of calls for each phone. In our simulations, we conservatively assume that phone call duration is an average of 3 minutes long [62].

[5]These small differences are due to our use of an additional month from the data set.

The results for a forming update location management policy are shown as the middle, green line in Figure 3. For a one-month sequence of locations — corresponding to changing IDs via ZipPhone once a month — accuracy falls by more than half, as compared to always-updating, to about 34%. For a one-hour sequence of cell transitions, accuracy is at 6%. As an impractical lower bound, we also show results for assigning new IMSI IDs after each cell transition. In that case, the attacker will succeed with less than 1% accuracy.

The bottom, blue line shows the lower bound for this inference technique at each trace duration: it represents an unrealistic location management scheme where the carrier learns only the location area but not the cell a user is associated with.

Limitations. We focus on the algorithm in Mulder et al. because it is a simple approach that is designed for GSM networks, and a universally accepted result. In future work, we plan to evaluate additional inference algorithms. The Eagle et al. data is relatively small; it has the advantage of containing cell IDs and call records for individual users (c.f., taxi cab data with no user-level consistency or call records). But we seek to expand our evaluation of ZipPhone to larger data sets. Finally, we hope to implement a version of ZipPhone using OpenBSC to understand the costs and limitations of frequent identifier updates. Additional attacks on ZipPhone are discussed in the following section.

VI. Discussion: Other Attacks

ZipPhone is vulnerable to at least two other major sets of attacks[6], which we discuss in this section: enumeration of all phones participating in ZipPhone; and active localization by carriers against phones in-between forming LA updates. In this preliminary work, we discuss each of these attacks, and leave empirical analysis for future work.

A. Identifying SIMs Participating in ZipPhone

In many but not all cases, a carrier can determine which SIMs on its network are running ZipPhone or the PeerPhone variant. Once identified, the carrier may elect to deny service to these users. In sum, the centralized protocol (e.g., an MVNO such as Zipline, running the ZipPhone protocol) is immune to most simple attacks, yet does not require the user trust Zipline.

First, the carrier can attempt to *enumerate* all SIM cards that are participating in the system. Enumeration is possible in PeerPhone if all venues where peers advertise their interest in lending credentials can be found and joined. This enumeration is not possible with an MVNO, unless it repeats IMSI values or colludes with a carrier.

Second, the carrier can *compare the latency* of a handset in responding to network signaling to others on the network. It should be significantly greater if a remote SIM is actually being contacted to complete authentication signaling. In the ZipPhone MVNO-based case, the values can be prefetched, defeating this attack; in PeerPhone exchange, they cannot be prefetched.

[6]Another risk we do not discuss is the physical danger of assuming another user's credentials [50].

Third, the carrier can *observe the locations* (i.e., cells and sectors) where a SIM card is being used and look for SIMs that follow a geographic pattern that is not typical, for example, jumping from city to city or country to country faster than a person can reasonably travel. PeerPhone might make this attack easier because the SIM will always return to its home location when a lending session terminates — unless the user only uses the SIMs of others via a fair exchange protocol. Zipline would not be subject to this attack since it has the option of never reusing an IMSI.

Fourth, a carrier can isolate those users that never receive a phone call and always use VOIP. In the MVNO case, again this analysis might not be effective if the IMSI is never used again, but would still raise suspicion depending on the length of time the IMSI is active in the network. As a defense, a user could place phone calls occasionally to arbitrary numbers (e.g., libraries or businesses) but we don't consider such a defense here.

Finally, whether in the peer-based or MVNO scenario, any phone that is engaging in the pageknocking defense against an active attacker described in Section IV-C will be fairly obvious.

B. Active Attacks

The active attacker's success depends on the method and quality of localization used, and the frequency with which they localize each user.

Most studies of cellular localization assume the phone itself participates in localization, which is more accurate (e.g., [13], [58]) but is not available against a ZipPhone user, according to our attacker model. Other studies assume that the phone is coerced into revealing its location [16], [21], [22], but we expect that such an attack can be thwarted by pageknocking, though we do not evaluate the defense here. Accordingly, active carriers have two main options:

Localization based on Cell ID. The carrier may localize the user based on the Cell ID it is associated with. A phone always knows its Cell ID, but the carrier learns the Cell ID of a phone only when it uses voice or data services. Cells themselves can vary greatly in size, from 100 km^2 to 1 km^2 (or smaller if dedicated to a single home or business). Trevisani and Vitaletti [57] performed in-field experiments in the US and found that Cell IDs are offset from the true location of the mobile by about 0.8 km in an urban environment (Manhattan), 0.5 km in a suburban environment, and 2.9 km in a highway environment. Watzdorf and Michahelles [59] report similar results. Ficek [22] reports accuracy of 0.05–20 km.

Localization based on multilateration. The carrier may use the *Uplink Time Difference of Arrival* (U-TDoA) method of multilateration. In U-TDoA, the carrier tracks the time it takes for the same signal to reach its network of base stations. The error is much lower than Cell ID at 0.11 km in tests [1] (Ficek [22] reports 0.04–0.12km). However, deploying U-TDoA requires a massive upgrade to the network infrastructure [22]. Accordingly, CDMA-based carriers, Verizon and Sprint, have instead deployed *assisted GPS (aGPS)*, which requires participation from the phone, and thus must use Cell IDs for passive localization. Only AT&T and T-Mobile have deployed U-TDoA. (Most European operators use Cell ID.)

In practice, there are limits on the carrier's ability to track all users via multilateration. Ficek [21] has shown some of these limits. For example, to locate a user via a false SMS message, a series of signals and the GSM broadcast paging channel are needed. These resources limit the number of users that can be tracked per minute to very low numbers depending on various parameters, e.g., tracking more than 20 users out of 410 exceeds network capacity. During the tracking, phone calls and data traffic that rely on the broadcast channel must be delayed.

In future work, we plan to evaluate the effectiveness of these two strategies and the cost to the carrier in terms of broadcast capacity.

VII. Related Work

There are hundreds of papers on location privacy [25], [33] for mobile users, spanning a number of paradigms from indoor mobile ad hoc networks to outdoor cellular networks, and from queries for information to friend-finding services and social network check-ins. Very broadly, the papers most related to our work fall within two key topics.

Location Based Services. First, papers on *privacy-preserving location-based services* (LBS) generally assume that the user is submitting queries for service. For example, the user may query for the nearest restaurant or gas station. Cellular networks are an LBS in that the user's query is a request for mobile voice or data *service* (rather than *content*) from a tower within radio range.

Not all solutions and analyses of privacy-preserving LBS are easily applied to the cellular scenario. Cellular users cannot introduce a trusted intermediary to obfuscate the mobile's position or introduce fake queries to the carrier [38]. Solutions that assume the user can control the level of granularity of their location are also not applicable [63].

Past work on deanonymization of private traces of mobile users assume the user's pseudonym is unchanged throughout the trace. A small amount of external information, such as the person's home or work address [28], can deanonymize an obfuscated trace [7], [8], [24], [32], [37], [41] given a consistent identifier. In contrast, we strive to change pseudonyms as often as network resources can support, which may be in minutes. Indeed, work by Zang and Bolot [65] shows that suitably anonymizing a trace of 25 million cellular users across 50 states (30 billion records total) requires only that users have the same pseudonym for no longer than a day. A day's duration is unsuitable for Zang and Bolot's goal of supporting researchers that wish to characterize the behavior of users over time (while maintaining their privacy). On the other hand, the result is promising for users seeking privacy, who might be able to change their pseudonyms much more frequently than once per day.

Others solutions can be adapted to our cellular scenario. A cellular user can use mix zones [8], [23], abstain from service [9], and introduce some types of false information [31], [53].

Location Privacy for Mobile Users. Secondly, many works address cellular location privacy. In contrast to our work,

most assume the carrier is willing to deploy changes. Some focus on enlisting a (trusted) carrier to protect against a third party [20], [26], [27], [35]. Reed et al. [49] propose privacy from the carrier using onion routing, but does not consider the direct connection that must be made to a tower. Federrath et al. [19] propose a similar scheme that prevents linkability of calls between two parties but omit critical details regarding authentication to the carrier. Fatemi et al. [18] propose an anonymous scheme for UMTS using identity-based encryption, but unlike our approach, that scheme involves the carrier in the cryptographic exchange; they enumerate the vulnerabilities of similar works [29], [44], [64], [67]. The closest work to ours is Kesdogan et al. [30], which proposes using a trusted third party to create pseudonyms for GSM users, but also routes all calls through that provider, which allows it to characterize the calling pattern and infer the caller.

Anonymous VOIP. The TORFone project (http://torfone.org/) has implemented a TCP-based voice-over-Tor system. (Others have examined the feasibility of real-time traffic over TCP [11], [12], [43], [66].) Our prior work has characterized the performance of a multi-hop/Tor-like UDP-based voice-over-IP, and shown it performs with sufficient quality of service despite the three-hop paths [36].

VIII. Conclusion

We proposed ZipPhone, a method for obtaining location privacy without the active cooperation of the carriers that control the cellular infrastructure. ZipPhone is backwards-compatible and designed to allow new IMSI identifiers to be re-issued relatively frequently over the air. A cooperative but untrusted MVNO can easily support ZipPhone by issuing ephemeral identities and session keys, and it is less vulnerable to many attacks that are possible against PeerPhone, a peer-based version of ZipPhone. PeerPhone can be deployed without the logistics of managing a relationship with an MNO, but is more vulnerable to attacks.

We also propose a method of thwarting attacks that prod the user to associate with cells based on fake pages. Our solution is an application of portknocking.

Our empirical analysis of the ZipPhone's protection against passive attacks by a carrier points out the limitations of results in a previous study [41]. That work assumed an always-update policy which is too costly for carriers to implement in practice. Our simulation of a forming LA update shows that ZipPhone users that update identifiers hourly can expect to be deanonymized only 6% of the time, which is less than a sixth of the rate reported by previous work.

In future work, we plan to quantify ZipPhone's protection against other inference algorithms, using additional data sets. We also plan to quantify the success of attacks that determine which phones are using ZipPhone (or page knocking defenses). And we plan to quantify the success of active attackers that localize the user without the user's participation, and quantify the costs to the carrier for launching such attacks.

Acknowledgements. This work was supported in part by NSF award CNS-0905349.

References

[1] 3GPP. TS 45.811: Feasibility Study on Uplink TDOA in GSM and GPRS. http://www.3gpp.org/DynaReport/45811.htm, July 2007.

[2] 3GPP TS 04.08 version 7.21.0. Mobile radio interface layer 3 specification. http://www.etsi.org/deliver/etsi_ts/100900_100999/100940/07.21.00_60/ts_100940v072100p.pdf Default time values on page 574.

[3] 3GPP TS 23.060 version 11.7.0. Universal Mobile Telecommunications System (UMTS); General Packet Radio Service (GPRS); Service description. http://www.etsi.org/deliver/etsi_ts/123000_123099/123060/11.07.00_60/ts_123060v110700p.pdf.

[4] Advanced Surveillance Technology, Inc. Wide-area (spiderweb) passive interception system. media.wix.com/ugd/70bfc5_619d9e6fc11df018f9af893328e798dd.pdf.

[5] E. Barkan, E. Biham, and N. Keller. Instant Ciphertext-Only Cryptanalysis of GSM Encrypted Communication. *Journal of Cryptology*, 21(3):392–429, Mar. 2008.

[6] M. Bellare, R. Canetti, and H. Krawczyk. Keying Hash Functions for Message Authentication. In *Proc. Advances in Cryptology*, volume 1109 of *LNCS*, pages 1–15. 1996.

[7] A. Beresford and F. Stajano. Location privacy in pervasive computing. *IEEE Pervasive Computing*, 2(1):46–55, 2003.

[8] A. Beresford and F. Stajano. Mix zones: user privacy in location-aware services. In *Proc. Pervasive Computing and Communications Wrkshps*, pages 127–131, 2004.

[9] L. Bindschaedler, M. Jadliwala, I. Bilogrevic, I. Aad, J.-P. Hubaux, V. Niemi, and P. Ginzboorg. Track Me If You Can: On the Effectiveness of Context-based Identifier Changes in Deployed Mobile Networks. In *Proc. ISOC NDSS*, Feb. 2012.

[10] A. Biryukov, A. Shamir, and D. Wagner. Real Time Cryptanalysis of A5/1 on a PC. In *Proc. Intl. Wrkshp on Fast Software Encryption*, pages 1–18, 2001.

[11] E. Brosh, S. Baset, V. Misra, D. Rubenstein, and H. Schulzrinne. The Delay-Friendliness of TCP for Real-Time Traffic. *IEEE/ACM Trans. on Networking*, 18(5):1478–1491, 2010.

[12] E. Brosh, S. A. Baset, D. Rubenstein, and H. Schulzrinne. The delay-friendliness of tcp. In *Proc. ACM SIGMETRICS*, pages 49–60, 2008.

[13] M. Y. Chen, T. Sohn, D. Chmelev, D. Haehnel, J. Hightower, J. Hughes, A. LaMarca, F. Potter, I. Smith, and A. Varshavsky. Practical Metropolitan-Scale Positioning for GSM Phones. In *Proc. UbiComp*, volume 4206 of *LNCS*, pages 225–242, 2006.

[14] R. deGraaf, J. Aycock, and M. J. Jacobson. Improved Port Knocking with Strong Authentication. *Proc. Annual Computer Security Applications Conf.*, pages 451–462, 2005.

[15] S. Dey, N. Roy, W. Xu, R. R. Choudhury, and S. Nelakuditi. AccelPrint: Imperfections of Accelerometers Make Smartphones Trackable. In *Proc. ISOC NDSS*, Feb. 2014.

[16] K. Dufková, M. Ficek, L. Kencl, J. Novak, J. Kouba, I. Gregor, and J. Danihelka. Active GSM Cell-id Tracking: "Where Did You Disappear?". In *Proc. ACM Intl. Wrkshp on Mobile Entity Localization and Tracking in GPS-less Environments*, pages 7–12, 2008.

[17] N. Eagle and A. Pentland. Reality Mining: Sensing Complex Social Systems. *Personal and Ubiquitous Computing*, 10(4):255–268, 2006.

[18] M. Fatemi, S. Salimi, and A. Salahi. Anonymous roaming in universal mobile telecommunication system mobile networks. *IET Information Security Journal*, 4(2):93–103, 2010.

[19] H. Federrath, A. Jerichow, D. Kesdogan, and A. Pfitzmann. Security in Public Mobile Communication Networks. In *Proc. IFIP/TC6 Personal Wireless Communications*, pages 105–116, April 1995.

[20] H. Federrath, A. Jerichow, and A. Pfitzmann. MIXes in Mobile Communication Systems: Location Management with Privacy. In *Proc. Intl. Wrkshp on Information Hiding*, pages 121–135, 1996.

[21] M. Ficek. *Tracking Users in Mobile Networks: Data Acquisition Methods and their Limits*. PhD thesis, Czech Technical University in Prague, https://dspace.cvut.cz/bitstream/handle/10467/18881/

TEZE_Diserta%C4%8Dn%C3%AD%20pr%C3%A1ce_Ficek_Michal_2013.pdf, June 2013.

[22] M. Ficek, T. Pop, and L. Kencl. Active tracking in mobile networks: An in-depth view. *Computer Networks*, 57(9):1936 – 1954, 2013.

[23] J. Freudiger, R. Shokri, and J.-P. Hubaux. On the Optimal Placement of Mix Zones. In *Proc. PETS*, pages 216–234, Aug. 2009.

[24] P. Golle and K. Partridge. On the anonymity of home/work location pairs. In *Proc. Intl. Conf. on Pervasive Computing*, pages 390–397, 2009.

[25] A. Görlach, A. Heinemann, and W. Terpstra. Survey on Location Privacy in Pervasive Computing. In *Privacy, Security and Trust within the Context of Pervasive Computing*, volume 780 of *The Intl. Series in Engineering and Computer Science*, pages 23–34. 2005.

[26] M. Gorlatova, R. Aiello, and S. Mangold. Managing base station location privacy. In *Proc. MILCOM*, pages 1201–1206, Nov. 2011.

[27] M. Gorlatova, R. Aiello, and S. Mangold. Managing location privacy in cellular networks with femtocell deployments. In *Proc. WiOpt Symposium*, pages 418–422, May 2011.

[28] S. Isaacman, R. Becker, R. Cáceres, S. Kobourov, M. Martonosi, J. Rowland, and A. Varshavsky. Identifying Important Places in People's Lives from Cellular Network Data. In *Proc. Intl. Conf. on Pervasive Computing*, pages 133–151, 2011.

[29] Y. Jiang, C. Lin, X. Shen, and M. Shi. Mutual Authentication and Key Exchange Protocols for Roaming Services in Wireless Mobile Networks. *IEEE Trans. on Wireless Communications*, 5(9):2569–2577, 2006.

[30] D. Kesdogan, H. Federrath, A. Jerichow, and A. Pfitzmann. Location Management Strategies Increasing Privacy in Mobile Communication. In *Information Systems Security*, pages 39–48. 1996.

[31] H. Kido, Y. Yanagisawa, and T. Satoh. An anonymous communication technique using dummies for location-based services. In *Proc. Intl. Conf. on Pervasive Services*, pages 88–97, 2005.

[32] J. Krumm. Inference Attacks on Location Tracks. In *Proc. Intl. Conf. on Pervasive Computing*, pages 127–143, May 2007.

[33] J. Krumm. A survey of computational location privacy. *Personal Ubiquitous Comput.*, 13(6):391–399, Aug. 2009.

[34] M. Krzywinski. Port Knocking: Network Authentication Across Closed Ports. In *SysAdmin Magazine*, volume 12, pages 12–17. 2003.

[35] D. F. Kune, J. Koelndorfer, N. Hopper, and Y. Kim. Location leaks on the GSM Air Interface. In *Proc. ISOC NDSS*, Feb. 2012.

[36] M. Liberatore, B. Gurung, B. N. Levine, and M. Wright. Empirical Tests of Anonymous Voice Over IP. *Elsevier Journal of Network and Computer Applications*, 34(1):341–350, January 2011.

[37] C. Y. Ma, D. K. Yau, N. K. Yip, and N. S. Rao. Privacy vulnerability of published anonymous mobility traces. In *Proc. MobiCom*, pages 185–196, 2010.

[38] J. Meyerowitz and R. Roy Choudhury. Hiding Stars with Fireworks: Location Privacy Through Camouflage. In *Proc. Annual Intl. Conf. on Mobile Computing and Networking*, pages 345–356, 2009.

[39] I. Miers, C. Garman, M. Green, and A. D. Rubin. Zerocoin: Anonymous Distributed E-Cash from Bitcoin. In *Proc. IEEE Symposium on Security and Privacy*, pages 397–411, 2013.

[40] Mobile Europe. Deutsche telekom upgrades mobile network security with a5/3 encryption standard. http://www.mobileeurope.co.uk/Press-Wire/deutsche-telekom-upgrades-mobile-network-security-with-a5-3-encryption-standard, December 2013.

[41] Y. D. Mulder, G. Danezis, L. Batina, and B. Preneel. Identification via Location-profiling in GSM Networks. In *Proc. ACM Wrkshp on Privacy in the Electronic Society*, pages 23–32, 2008.

[42] S. Nakamoto. Bitcoin: A Peer-to-Peer Electronic Cash System. http://www.bitcoin.org/bitcoin.pdf, 2009.

[43] M. F. Nowlan, N. Tiwari, J. Iyengar, S. O. Aminy, and B. Fordy. Fitting square pegs through round pipes: Unordered delivery wire-compatible with tcp and tls. In *Proc. USENIX NSDI*, pages 28–28, 2012.

[44] J. Park, J. Go, and K. Kim. Wireless authentication protocol preserving user anonymity. In *Proc. SCIS*, pages 159–164, 2001.

[45] A. Polak, S. Dolathshahi, and D. Goeckel. Identifying Wireless Users via Transmitter Imperfections. *IEEE JSAC: Special Issue on Advances in Digital Forensics for Communications and Networking*, 29(7):1469–1479, August 2011.

[46] A. Polak and D. Goeckel. RF Fingerprinting of Users who Actively Mask Their Identities with Artificial Distortion. In *Proc. Asilomar Conf. on Signals, Systems, and Computers*, May 2011.

[47] J. Rao, P. Rohatgi, H. Scherzer, and S. Tinguely. Partitioning attacks: or how to rapidly clone some GSM cards. In *IEEE Symp. Security and Privacy*, pages 31–41, 2002.

[48] S. M. Razavi. *Tracking Area Planning in Cellular Networks [Elektronisk resurs] : Optimization and Performance Evaluation*. Linköping, 2011.

[49] M. G. Reed, P. F. Syverson, and D. M. Goldschlag. Protocols using anonymous connections: Mobile applications. In *Security Protocols*, volume 1361 of *LNCS*, pages 13–23. 1998.

[50] J. Scahill and G. Greenwald. The NSA's Secret Role in the U.S. Assassination Program. https://firstlook.org/theintercept/article/2014/02/10/the-nsas-secret-role/, 10 Feb 2014.

[51] B. Schneier. The eternal value of privacy. *Wired*, May 2006.

[52] Security Research Labs. Gsm map project: gsm security country report (usa). http://gsmmap.org/assets/pdfs/gsmmap.org-country_report-United_States_of_America-2013-08.pdf, August 2013.

[53] R. Shokri, G. Theodorakopoulos, G. Danezis, J.-P. Hubaux, and J.-Y. Boudec. Quantifying Location Privacy: The Case of Sporadic Location Exposure. In *Proc. PETS*, pages 57–76, Aug. 2011.

[54] C. Smith and D. Collins. *3G Wireless Networks*. 2nd edition, 2007.

[55] H. Soroush, K. Sung, E. Learned-Miller, B. N. Levine, and M. Liberatore. Disabling GPS is Not Enough: Cellular location leaks over the Internet. In *Proc. PETS*, pages 103–122, July 2013.

[56] C. Timberg and A. Soltani. By cracking cellphone code, NSA has capacity for decoding private conversations. http://www.washingtonpost.com/business/technology/by-cracking-cellphone-code-nsa-has-capacity-for-decoding-private-conversations/2013/12/13/e119b598-612f-11e3-bf45-61f69f54fc5f_story.html, December 13 2013.

[57] E. Trevisani and A. Vitaletti. Cell-ID Location Technique, Limits and Benefits: An Experimental Study. In *Proc. IEEE Wrkshp on Mobile Computing Systems and Applications*, pages 51–60, 2004.

[58] A. Varshavsky, M. Chen, E. de Lara, J. Froehlich, D. Haehnel, J. Hightower, A. LaMarca, F. Potter, T. Sohn, K. Tang, and I. Smith. Are GSM Phones THE Solution for Localization? In *Proc. Wrkshp Mobile Computing Systems and Applications*, pages 34–42, 2006.

[59] S. von Watzdorf and F. Michahelles. Accuracy of Positioning Data on Smartphones. In *Proc. Intl. Wrkshp on Location and the Web*, pages 2:1–2:4, 2010.

[60] Wikipedia. List of us mobile mvnos. Last checked 2013-12-31. http://en.wikipedia.org/wiki/List_of_United_States_mobile_virtual_network_operators.

[61] V. W.-S. Wong and V. C. Leung. Location Management for Next-generation Personal Communications Networks. *IEEE Network*, 14(5):18–24, Sept. 2000.

[62] J. Wortham. Cellphones Now Used More for Data Than for Calls. http://www.nytimes.com/2010/05/14/technology/personaltech/14talk.html, May 13 2010.

[63] T. Xu and Y. Cai. Feeling-based Location Privacy Protection for Location-based Services. In *Proc. ACM CCS*, pages 348–357, 2009.

[64] G. Yang, D. Wong, and X. Deng. Efficient anonymous roaming and its security analysis. In *Applied Cryptography and Network Security*, volume 3531 of *LNCS*, pages 334–349. 2005.

[65] H. Zang and J. Bolot. Anonymization of Location Data Does Not Work: A Large-scale Measurement Study. In *Proc. ACM MobiCom*, pages 145–156, 2011.

[66] X. Zhang and H. Schulzrinne. Voice over tcp and udp. Technical Report CUCS-033-04, Columbia University, 2004.

[67] J. Zhu and J. Ma. A new authentication scheme with anonymity for wireless environments. *IEEE Trans. on Consumer Electronics*, 50(1):231–235, 2004.